AMENDMENTS TO THE CLAIMS

Please amend the claims to read as follows:

1. (Currently Amended) A method for synchronous debugging of a parallel processing platform, the platform comprising a plurality of processors executing code, the code including one or more breakpoints to allow debugging of the code, the method comprising:

   upon a processor reaching a breakpoint, propagating ~~a halt~~ an interrupt command to a standard interrupt pin of all of the processors in the platform;

   thereby halting system execution synchronously to enable examination of the states of the processors.

2. (Original) The method of claim 1 further including, upon receipt of a resume command, propagating the resume command to all the processors in the platform thereby enabling synchronous restart of the execution of the code by the processors.

3. (Original) The method of claim 1 wherein the propagating of the halt command to all of the processors in the platform comprises:

   a. the processor that reached the breakpoint generating an interrupt output signal to a hardware I/O device;

   b. the hardware I/O device propagating the interrupt output signal to all the processors in the platform.

4. (Currently Amended) The method of claim 1 wherein the processors are grouped in clusters, with each cluster including an interrupt controller and a register, the clusters are grouped in modules, with each module including an OR gate driver, the OR gate drivers are connected via a platform backplane, and wherein the propagating of the halt command to all of the processors in the platform comprises:

a. gencrating by the processor that reached the breakpoint ~~generating~~ an output signal to the register of its cluster;

b. gencrating by the register ~~generating~~ an output command signal to the OR gate driver of its module;

c. gencrating by the OR gate driver ~~generating~~ an output command signal via the backplane to the other OR gate drivers in the platform;

d. gencrating by each OR gate driver ~~generating~~ an output command signal to the interrupt controller of each cluster in the OR gate driver's module;

e. generating by each interrupt controller of each cluster ~~generating~~ an output signal to an interrupt pin on each processor in the cluster;

thereby causing the processors to halt execution.

5. (Currently Amended) A parallel processing system for synchronous debugging of a parallel processing platform, the platform comprising a plurality of processors executing code, the code including one or more breakpoints to allow debugging of the code, the system comprising:

electrical circuitry for propagating, upon a processor in the platform reaching a breakpoint, ~~a halt signal~~ an interrupt command to a standard interrupt pin of all the processors in the system;

thereby halting system execution synchronously to enable examination of the states of the processors.

6. (Original) The system of claim 5 wherein said electrical circuitry upon receiving a resume command, propagates the resume command to all the processors in the system, thereby resuming system execution synchronously.

7. (Original) The system of claim 5, wherein the electrical circuitry for propagation comprises:

a. each processor being connected to a hardware I/O device;

b. each hardware I/O device including an output signal pin;

c. each output signal pin connected via OR gate drivers to interrupt pins on every processor in the system.

8. (Original) The system of claim 5, the processors grouped into one or more clusters, the clusters grouped into one or more modules, each module including an OR gate driver, the OR gate drivers connected via a platform backplane, the electrical circuitry for propagation comprising a system controller per cluster; the system controller comprising a register and an interrupt controller; the register including a halt/resume command signal output pin; the command signal output pins of all the registers in a module connected to the module's OR gate driver; the OR-gate driver replicating a command signal at any of its inputs to its outputs; the OR-gate driver having outputs to the other OR-gates in the platform and to the interrupt controller on each cluster in the OR gate driver's module; each interrupt controller having an output connected to an interrupt pin on each processor in the interrupt controller's cluster.